

# **HDF to GeoTIFF Science Processing Algorithm (H<sub>2</sub>G\_SPA)**

## **User's Guide**

**Version 1.6a**

April 2010



---

**GODDARD SPACE FLIGHT CENTER  
GREENBELT, MARYLAND**

## Table of Contents

1	General.....	1
2	Algorithm Wrapper Concept.....	1
3	Software Description.....	1
4	Software Version .....	2
5	Credits .....	3
6	Prerequisites.....	3
7	Program Inputs and Outputs.....	3
8	Installation and Configuration .....	3
8.1	Installing as a Standalone Application .....	3
8.2	Installing into an IPOPP Framework .....	4
9	Software Package Testing and Validation .....	5
10	Program Operation.....	5
10.1	To Use the Run Scripts .....	6
10.2	To Use the Scripts in the Testscripts Directory.....	8
11	H <sub>2</sub> G Image Products .....	9
11.1	Standard Products.....	9
11.1.1	Generating H <sub>2</sub> G Standard Products .....	12
11.1.2	Overriding Projection and Resolution of Standard Products .....	13
11.1.3	Subsetting H <sub>2</sub> G Standard Products.....	15
11.2	User-defined Products.....	15
11.2.1	Configuration Files for User-defined Products .....	16
11.2.2	Overriding Projection and Resolution in the User-defined Configuration File .....	24
11.2.3	Subsetting User-defined Products .....	25
	Appendix A. Dataset Identifiers.....	A-1
	Appendix B. Geolocation Identifiers .....	B-1
	Appendix C. Modifying Maximum Java Heap Size.....	C-1
	Appendix D. H <sub>2</sub> G Standard Product Descriptions .....	D-1

## **1 General**

The NASA Goddard Space Flight Center's (GSFC) Direct Readout Laboratory (DRL), Code 606.3 developed this wrapper software for the National Polar-orbiting Operational Environmental Satellite System (NPOESS) Preparatory Project (NPP) In-Situ Ground System (NISGS) and the International Polar Orbiter Processing Package (IPOPP).

Users must agree to all terms and conditions in the Software Usage Agreement on the DRL Web Portal before downloading this software.

Software and documentation published on the DRL Web Portal may occasionally be updated or modified. The most current versions of DRL software are available at the DRL Web Portal:

<http://directreadout.sci.gsfc.nasa.gov>

Questions relating to the contents or status of this software and its documentation should be addressed to the DRL via the Contact Us mechanism at the DRL Web Portal:

<http://directreadout.sci.gsfc.nasa.gov/index.cfm?section=contact%20us>

## **2 Algorithm Wrapper Concept**

The DRL has developed an algorithm wrapper to provide a common command and execution interface to encapsulate multi-discipline, multi-mission science processing algorithms. The wrapper also provides a structured, standardized technique for packaging new or updated algorithms with minimal effort.

A Science Processing Algorithm (SPA) is defined as an algorithm to which the wrapper has been applied. SPAs will function in a standalone, cross-platform environment to serve the needs of the broad Direct Readout community. Detailed information about SPAs and other DRL technologies is available at:

<http://directreadout.sci.gsfc.nasa.gov/index.cfm?section=technology>

## **3 Software Description**

This DRL software package contains the H<sub>2</sub>G\_SPA (Hierarchical Data Format [HDF] to Georeferenced Tagged Image File Format [GeoTIFF] Converter Science Processing Algorithm). H<sub>2</sub>G\_SPA is specially designed for Direct Readout and can create geolocated GeoTIFF images, jpeg browse images, and png browse images for various parameter datasets in Level 2 MODIS SPA products. H<sub>2</sub>G also creates standard true color images and user-defined false color images from Level 2

Corrected Reflectance Products. The H<sub>2</sub>G\_SPA functions in two modes: Standalone, or as an IPOPP plug-in.

The geolocated GeoTIFF images are Geographic Information System (GIS)-ingestible and can also be opened by standard image viewers. The non-geolocated jpeg and png images are more suitable as browse images. These browse images are enhanced with vector overlays of land/sea and political boundaries.

H<sub>2</sub>G\_SPA incorporates features to enhance output images and facilitate scientific interpretation:

- while creating images from a primary dataset, a secondary dataset may be used to mask appropriate areas;
- user-defined color map and user-defined scaling capabilities for conversion of dataset values into image pixels;
- choice of either geographic or stereographic projection for the output image;
- optional subsetting of swaths into user-defined regions of interest (currently available for geographically-projected images only);
- jpeg and png browse images can have legends and/or fire pixel overlays.

H<sub>2</sub>G currently allows the user to select geographic and stereographic projections. Inclusion of other projections and image mosaic capabilities is under consideration for future releases of H<sub>2</sub>G\_SPA.

For more details about how the H<sub>2</sub>G\_SPA software creates images, go to:

<http://directreadout.sci.gsfc.nasa.gov/index.cfm?section=technology&page=NISGS&subpage=H2G/overview>

## 4 Software Version

Version 1.1 of the DRL algorithm wrapper was used to package the SPA described in this document. The SPA uses H<sub>2</sub>G processing code (Alpha Version 1.6a, February 2010). The H<sub>2</sub>G\_SPA stereographic projection capability utilizes the JPROJ.4 Java Native Interface (JNI) to the PROJ.4 Cartographic Projections Library. This library was initially developed by the U.S. Geological Survey (USGS) and is currently being maintained/enhanced by the Open Source Geospatial Foundation (OSGeo).

Enhancements to Version 1.6a include:

- capability to produce stereographically projected images;
- allowing projection and resolution parameters on the command line to override defaults set in standard or user-defined configuration files;
- allowing subsetting capabilities on the command line (currently available for geographically-projected images only);
- addition of the Cohen-Sutherland Line Clipping algorithm for proper vector overlays;
- removal of artifacts in RGB images at  $0^{\circ}$  longitude.

H<sub>2</sub>G\_SPA runs on 64-bit Linux platforms only. This package has been tested successfully on the following platforms: Fedora 10, CentOS 5.3, Kubuntu 8.10, and SUSE 11.1.

Copyright 1999-2007, United States Government as represented by the Administrator for the National Aeronautics and Space Administration. All Rights Reserved.

## 5 Credits

H<sub>2</sub>G was developed by the DRL at NASA/GSFC.

## 6 Prerequisites

To run this package, you must have the Java Development Kit (JDK) or Java Runtime Engine (JRE) (Java 1.5 or higher) installed on your computer, and the bin directory of your Java installation in your PATH environment variable.

## 7 Program Inputs and Outputs

H<sub>2</sub>G\_SPA can take most Level 2 MODIS SPA products as input. H<sub>2</sub>G\_SPA has been pre-configured to produce a number of standard Level 2 image products. However, users can configure H<sub>2</sub>G\_SPA to produce non-standard image products.

## 8 Installation and Configuration

H<sub>2</sub>G\_SPA can be run either as a standalone application or can be installed dynamically into an IPOPP framework as a plug-in.

### 8.1 Installing as a Standalone Application

Download the H2G\_1.6a\_SPA\_1.1.tar.gz and H2G\_1.6a\_SPA\_1.1\_testdata.tar.gz (optional) files into the same directory.

Decompress and un-archive the H2G\_1.6a\_SPA\_1.1.tar.gz and H2G\_1.6a\_SPA\_1.1\_testdata.tar.gz (optional) files:

```
$ tar -xzf H2G_1.6a_SPA_1.1.tar.gz  
$ tar -xzf H2G_1.6a_SPA_1.1_testdata.tar.gz
```

This will create the following subdirectories:

```
SPA  
h2g  
    algorithm  
    station  
    wrapper  
    ancillary  
    testscripts  
    testdata
```

H<sub>2</sub>G\_SPA was compiled with Java 1.5. H<sub>2</sub>G\_SPA is configured to run with a maximum Java heap size of 2GB, which can cause H<sub>2</sub>G\_SPA to fail on a computer with inadequate memory. Users may increase or decrease the memory specifications. Refer to Appendix C for instructions.

**NOTE:** Examples supplied from this point forward assume that the SPA was installed into /home/ipopp/drl, although in Standalone mode SPAs may be installed into any directory.

## 8.2 Installing into an IPOPP Framework

H<sub>2</sub>G\_SPA can be installed dynamically into an IPOPP framework. Instructions are contained in the IPOPP User's Guide. H<sub>2</sub>G\_SPA is pre-configured to produce geographically projected GeoTIFFs in IPOPP Mode. Users can configure H<sub>2</sub>G\_SPA in IPOPP Mode to produce stereographically projected GeoTIFFs instead by performing an optional configuration step prior to issuing the NISGSinstall.sh command, as follows:

cd into the algorithm directory and execute the configure-ipopp-projection script. Input '2' when asked for the Projection options for IPOPP Mode, as follows:

```
$ cd /home/ipopp/drl/SPA/h2g/algorithm  
$ ./configure-ipopp-projection  
Projection Options for IPOPP Mode: Select One  
1. Geographic 2. Stereographic  
2  
Configuring h2g stations for Stereographic projections
```

H<sub>2</sub>G\_SPA is now ready to be installed into the IPOPP framework as described in the IPOPP User's Guide.

**NOTE:** The configure-ipopp projection script enables the user to reconfigure H<sub>2</sub>G\_SPA for geographic projections in IPOPP Mode. However, once reconfiguration is complete, the user must reinstall H<sub>2</sub>G\_SPA into the IPOPP Framework as instructed in the IPOPP User's Guide.

## 9 Software Package Testing and Validation

The testscripts subdirectory contains test scripts that can be used to verify that your current installation of the SPA is working properly, as described below. Note that the optional H2G\_1.6a\_SPA\_1.1testdata.tar.gz file is required to execute these testing procedures.

*Step 1:* cd into the testscripts directory.

*Step 2:* Run the 'run-lst-tiff' script by typing:

```
$./run-lst-tiff
```

A successful execution usually takes some time, so if the execution seems to get stuck, do not become impatient. If everything is working properly, the script will terminate with a message such as:

```
Output modis.h2gout is /home/ipopp/drl/SPA/h2g/testdata/output/LST.tif
```

You can cd to the output directory to verify that the science product (in this example the Land Surface Temperature [LST] GeoTIFF image) exists. If it does exist, then the process executed successfully. If there is a problem and the code terminates abnormally, the problem can be identified using the log files. Log files are automatically generated within the directory used for execution. They start with stdfile\* and errfile\*. Please report any errors that cannot be fixed to the DRL. Test output file(s) are provided for comparison in the testdata/output/h2g\_standard\_outputs directory. The output products serve as an indicator of expected program output. Use a comparison utility (such as a standard image viewer) to compare your output(s) to those provided in the testdata/output/h2g\_standard\_outputs subdirectory.

## 10 Program Operation

In order to run the package using your own input data, you can either use the run scripts within the wrapper directory, or modify the test scripts within the testscripts directory.

## 10.1 To Use the Run Scripts

**Identify the 'run' script:** The wrapper directory within this package contains the h2g subdirectory. You must execute the 'run' within the h2g/wrapper/h2g subdirectory to execute H<sub>2</sub>G\_SPA. Note that to execute 'run', you need to have java on your path.

**Specify input parameters using <label value> pairs:** To execute the 'run' script, you must supply the required input and output parameters. Input and output parameters are usually file paths or other values (e.g., the output type). Each parameter is specified on the command line by a <label value> pair. Labels are simply predefined names for parameters. Each label must be followed by its actual value. The <label value> pairs must be specified on the command line in order for H<sub>2</sub>G\_SPA to execute. Some of these pairs are optional, meaning the process would still be able to execute even if that parameter is not supplied. There are three types of <label value> pairs that the H<sub>2</sub>G\_SPA uses, as follows:

- a) Input file label/values. These are input file paths. Values are absolute or relative paths to the corresponding input file.
- b) Parameter label/values. These are parameters that need to be passed onto the SPA (e.g., the image output type).
- c) Output file labels. These are output files that are produced by the SPA. Values are the relative/absolute paths of the files you want to generate.

Table 1 contains labels, and their brief descriptions required by H<sub>2</sub>G. Section 11, "H<sub>2</sub>G Image Products," contains detailed descriptions and examples of usage of these labels.

**Table 1. Labels and Descriptions Required by H<sub>2</sub>G**

Input File Labels	Description
modis.data	Path to the MODIS L2 HDF file that contains the parameter dataset for which the image is being created.
modis.mask (optional)	Path to the MODIS L2 HDF file that contains the mask dataset used as a mask for the image being created. modis.mask is not needed if either no mask is being used or the mask dataset is contained within modis.data.
modis.geo (optional)	Path to the MOD03 geolocation HDF file or the L2 HDF file which contains the latitude and longitude datasets for the same swath. modis.geo is not needed only when the geolocation information is within the modis.data file (e.g., HDF outputs from IMAPP_SPA have their own geolocation).
modis.fireloc (optional)	Path to the fire-location text file. The fire-location text file is produced by the MOD14_SPA. modis.fireloc can be used to overlay fire pixels on the output image. modis.fireloc can be used only when the image output type is geotiff.rgb, jpeg.rgb or png.rgb. (See output.type label description below.) The configuration file must contain the FIRESOURCE keyword for fire pixel overlays to work.
Output File Label	Description
modis.h2gout	Path to the geotiff, jpeg or png image product generated by H <sub>2</sub> G.
Parameter Label	Description
config.type	Configuration type. Can be (a) 'standard' for the standard H <sub>2</sub> G products; (b) 'singleband' for a user-defined single band image; or (c) 'rgb' for a user-defined RGB image.
config.name	config.name is either (a) the identifier for the standard H <sub>2</sub> G products; or (b) the path to the user-defined configuration file.
output.type	<p>output.type can be either (a) geotiff.u8cm (for an 8-bit colormap embedded GeoTIFF image); (b) geotiff.rgb (for an RGB GeoTIFF image); (c) jpeg.rgb (for an RGB jpeg image); or (d) png.rgb (for an RGB png image).</p> <p><b>NOTE:</b> jpeg.rgb and png.rgb images do not have geolocation information. They are more useful as browse images. jpeg.rgb and png.rgb images have vector overlays of land/sea/political boundaries and may have legends.</p>
projection (optional)	Projection can be either (a) geographic or (b) stereographic. This parameter is used to override the projection defined for the H <sub>2</sub> G standard product or in the user-defined configuration file.

resolution (optional)	Used to specify the resolution of the output image product. When using the geographic projection, resolution is in degrees. For the stereographic projection resolution, is in meters.
Input File Labels	Description
toplat botlat leftlon rightlon (optional)	These 4 parameters corresponding to top latitude, bottom latitude, left longitude, right longitude can be used together to specify a subset of the whole granule. (Subsetting is not available for stereographic projections.)

**NOTE:** The modis.data, modis.geo and modis.mask HDF and modis.fireloc text files must correspond to the same swath.

## 10.2 To Use the Scripts in the Testscripts Directory

One simple way to run the algorithms from any directory of your choice, using your own data, is to copy the corresponding run-xxx scripts from the testscripts directory and its subdirectories (more\_examples and other\_examples) to your selected directory. Change the values of the different variables to reflect the file paths of the wrapper directories and the input/output files. Then modify the input/output file name variables. If required, add more parameters to the command line. Run the scripts to process your data.

## 11 H<sub>2</sub>G Image Products

This section describes:

- *What are H<sub>2</sub>G standard products?*
  - *How to generate the standard products?*
  - *How to override the projection and resolution of the standard products?*
  - *How to subset the standard products?*
- *What are H<sub>2</sub>G user-defined products?*
  - *How to write configuration files for user-defined products?*
  - *How to override the projection and resolution defined in the configuration file?*
  - *How to subset user-defined products?*

H<sub>2</sub>G can produce two types of products: standard and user-defined. H<sub>2</sub>G is pre-configured to generate the standard image products contained in Table 2, as described below in "Standard Products." Generation of user-defined (i.e., non-standard) image products, however, requires the user to undertake the additional step of writing a unique configuration file. These configuration files are simple text files that supply the user configuration to H<sub>2</sub>G. Detailed instructions to generate standard and user-defined image products follow. Users are reminded that numerous examples of command-lines/configuration-files are contained within the H2G\_1.6a\_SPA\_1.1\_testdata.tar.gz archive.

### 11.1 Standard Products

The standard outputs can be generated by setting the config.type label to 'standard' and using the correct identifier for the config.name label. Table 2 contains a list of standard outputs, with corresponding config.name identifiers and required input parameters. H<sub>2</sub>G standard outputs are by default in geographic projection; instructions to override the default setting are contained in paragraph 11.1.2.

**Table 2. H<sub>2</sub>G Standard Outputs and Corresponding Input Requirements**

'config.name' Identifiers for H2G Standard Products	Parameter	Inputs
ndvi	NDVI (masks: water and cloud; Resolution: $0.01^{\circ}$ )	modis.data <mod13 HDF output from NDVIEVI_SPA> modis.mask <L2 HDF output from MOD14_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name ndvi output.type <geotiff.u8cm   jpeg.argb png.argb>
evi	EVI (masks: water and cloud; Resolution: $0.01^{\circ}$ )	modis.data <mod13 HDF output from NDVIEVI_SPA> modis.mask <L2 HDF output from MOD14_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name evi output.type <geotiff.u8cm   jpeg.argb png.argb >
lst	LST (masks: water and cloud; Resolution: $0.01^{\circ}$ )	modis.data <LST HDF output from MODLST_SPA> modis.mask <L2 HDF output from MOD14_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name lst output.type <geotiff.u8cm   jpeg.argb png.argb>
fire	Fire Mask (Resolution: $0.01$ )	modis.data <MOD14 HDF output from MOD14_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> modis.fireloc <firelocation text file output from MOD14_SPA; ignored when output.type is geotiff.u8cm> config.type standard config.name fire output.type <geotiff.u8cm   jpeg.argb png.argb >
sst	Sea Surface Temperature (Mask: SST quality flag; Resolution: $0.01^{\circ}$ )	modis.data <SST HDF output from L2GEN_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name sst output.type <geotiff.u8cm   jpeg.argb png.argb >
chlor	Chlorophyll-a concentration (Mask: l2flags; Resolution: $0.01^{\circ}$ )	modis.data <Chlor HDF output from L2GEN_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name chlor output.type <geotiff.u8cm   jpeg.argb png.argb >

aerosol-aod	Aerosol Optical Depth (Resolution: $0.1^{\circ}$ )	modis.data <mod04 HDF output from IMAPP_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name aerosol-aod output.type <geotiff.u8cm   jpeg.rgb png.rgb >
cloudtop-irphase	Cloud Phase (Resolution: $0.05^{\circ}$ )	modis.data <mod06 HDF output from IMAPP_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name cloudtop-irphase output.type <geotiff.u8cm   jpeg.rgb png.rgb >
cloudtop-ctp	Cloudtop Pressure (Resolution: $0.05^{\circ}$ )	modis.data <mod06 HDF output from IMAPP_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name cloudtop-ctp output.type <geotiff.u8cm   jpeg.rgb png.rgb >
atmprofile-tpw	Total Precipitable Water (Resolution: $0.05^{\circ}$ )	modis.data <mod07 HDF output from IMAPP_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name atmprofile-tpw output.type <geotiff.u8cm   jpeg.rgb png.rgb >
cloudmask	Cloudmask (Resolution: $0.01^{\circ}$ )	modis.data <mod35 HDF output from IMAPP_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name cloudmask output.type <geotiff.u8cm   jpeg.rgb png.rgb >
tcolor0_0025	True Color from corrected reflectances (Resolution: $0.0025^{\circ}$ )	modis.data <crefl HDF output from CREFL_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolor0_0025 output.type <geotiff.rgb   jpeg.rgb png.rgb >
tcolor0_005	True Color from corrected reflectances (Resolution: $0.005^{\circ}$ )	modis.data <crefl HDF output from CREFL_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolor0_005 output.type <geotiff.rgb   jpeg.rgb png.rgb >
tcolor0_01	True Color from corrected reflectances (Resolution: $0.01^{\circ}$ )	modis.data <crefl HDF output from CREFL_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolor0_01 output.type <geotiff.rgb   jpeg.rgb png.rgb >
tcolorfire0_0025	True Color with fire pixel overlays from corrected reflectances (Resolution: $0.0025^{\circ}$ )	modis.data <crefl HDF output from CREFL_SPA> modis.fireloc <firelocation text file output from MOD14_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolorfire0_0025 output.type <geotiff.rgb   jpeg.rgb png.rgb >

tcolorfire0_005	True Color with fire pixel overlays from corrected reflectances (Resolution: 0.005°)	modis.data <crefl HDF output from CREFL_SPA> modis.fireloc <firelocation text file output from MOD14_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolorfire0_005 output.type <geotiff.rgb   jpeg.rgb png.rgb>
tcolorfire0_01	True Color with fire pixel overlays from corrected reflectances (Resolution: 0.01°)	modis.data <crefl HDF output from CREFL_SPA> modis.fireloc <firelocation text file output from MOD14_SPA> modis.geo <MOD03 HDF output from MODISL1DB_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolorfire0_01 output.type <geotiff.rgb   jpeg.rgb png.rgb>
mod09tcolor0_005	True Color from surface reflectances (Resolution: 0.005°)	modis.data <mod09 HDF output from MOD09_SPA> modis.h2gout <Path to tif, png or jpeg output image file> config.type standard config.name mod09tcolor0_005 output.type <geotiff.rgb   jpeg.rgb png.rgb>

### 11.1.1 Generating H<sub>2</sub>G Standard Products

Command line examples to generate standard H<sub>2</sub>G products from the testscripts directory are provided below.

#### Example 1: TrueColor 0.01 degree tif (run from the testscripts directory)

```
$./wrapper/h2g/run \
config.type standard \
config.name tcolor0_01 \
modis.data ../testdata/input/MYDcrefl.08085185938.hdf \
modis.geo ../testdata/input/MYD03.08085185938.hdf \
modis.h2gout ../testdata/output/TCOLOR0.01.tif \
output.type geotiff.rgb
```

#### Example 2: EVI 0.01 degree jpeg (run from the testscripts directory)

```
$ ./wrapper/h2g/run \
config.type standard \
config.name evi \
modis.data ../testdata/input/MYD013.08085185938.hdf \
modis.geo ../testdata/input/MYD03.08085185938.hdf \
modis.mask ../testdata/input/MYD014.08085185938.hdf \
modis.h2gout ../testdata/output/EVI.jpg \
output.type jpeg.rgb
```

A successful execution usually takes some time, so if the execution seems to get stuck, do not become impatient. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be

recognized). Correct the problem and run again. The problem can also be identified using the stdfile\* and errfile\* log files. Log files are automatically generated within the directory used for execution.

#### **NOTES:**

1. Command Line Examples for generating H<sub>2</sub>G standard outputs are provided in the run-h2g-standard-output file within the testscripts directory. All command lines are commented. Please uncomment the desired command line and then type \$./run-h2g-standard-input to execute the command. Some H<sub>2</sub>G standard output products are available for comparison in the testdata/output/h2g\_standard\_outputs subdirectory.
2. The tcolor0\_01/tcolorfire0\_01/ndvi/evi standard products are optimized for ~1km true color images. For higher resolution true color images use the tcolor0\_0025, tcolor0\_05, tcolorfire0\_0025, tcolorfire0\_05, ndvi0\_0025, evi0\_0025 products. Users are cautioned that processing of the higher resolution true-color products is resource-intensive and may fail if there is insufficient memory/processing power on your computer.
3. The IMAPP aerosol product may produce insufficient geolocation data at higher latitudes. H<sub>2</sub>G would fail to produce correct aerosol image products in such cases.
4. Please see Appendix D for additional information on standard products, such as: (a) how the products were scaled into 8-bit images; (b) how the masks were used; and (c) how to retrieve actual parameter values from the GeoTIFF images.

#### **11.1.2 Overriding Projection and Resolution of Standard Products**

H<sub>2</sub>G standard products are by default in geographic projection. In order to override this default projection and use any other projection (currently the only other projection is stereographic), you should use the projection and resolution parameter labels on the command line. Note that the geographic projection requires resolution in latitude/longitude degree units, while the stereographic projection requires resolution in meter units. You may also override only the resolution label to get a geographically projected image in a different resolution.

**Example 3: TrueColor Stereographic 1000 meter tif** (run from the testscripts directory)

```
$./wrapper/h2g/run \
  config.type standard \
  config.name tcolor0_01 \
  modis.data ../testdata/input/MYDcrefl.08085185938.hdf \
  modis.fireloc ../testdata/input/MYD014.08085185938.txt \
  modis.geo ../testdata/input/MYD03.08085185938.hdf \
  modis.h2gout ../testdata/output/TCOLOR0.01_stereo.tif \
  output.type geotiff.argb \
  projection stereographic \
  resolution 1000
```

**Example 4: EVI Stereographic 1000m png** (run from the testscripts directory)

```
$./wrapper/h2g/run \
  config.type standard \
  config.name evi \
  modis.data ../testdata/input/MYD013.08085185938.hdf \
  modis.geo ../testdata/input/MYD03.08085185938.hdf \
  modis.mask ../testdata/input/MYD014.08085185938.hdf \
  modis.h2gout ../testdata/output/EVI_stereo.png \
  output.type png.argb \
  projection stereographic \
  resolution 1000
```

**Example 5: EVI Geographic 0.05 degree thumbnail png** (run from the testscripts directory)

```
$ ./wrapper/h2g/run \
  config.type standard \
  config.name evi \
  modis.data ../testdata/input/MYD013.08085185938.hdf \
  modis.geo ../testdata/input/MYD03.08085185938.hdf \
  modis.mask ../testdata/input/MYD014.08085185938.hdf \
  modis.h2gout ../testdata/output/EVI_thumbnail.png \
  output.type png.argb \
  resolution 0.05
```

## **NOTES:**

1. The center of projection for the stereographic projection is automatically selected by H<sub>2</sub>G at approximately the midpoint of the image.
2. When you overrride the default projection, be sure to also specify the resolution label along with the projection label. Care should be taken while specifying the resolution, as the resolution units of the geographic (degrees) and stereographic (meters) projections are different.

### **11.1.3 Subsetting H<sub>2</sub>G Standard Products**

Use the following parameter labels on the command line to specify a region of interest:

- a) botlat (specifies bottom latitude);
- b) toplat (specifies top latitude);
- c) leftlon (specifies left longitude);
- d) rightlon (specifies right longitude).

**NOTE:** Subsetting is currently available for the geographic projection only. If any of these parameters are specified for the stereographic projection, they will be ignored.

**Example 6: Subsetted TrueColor Geographic 0.01 degree tif** (run from the testscripts directory)

```
$./wrapper/h2g/run \
config.type standard \
config.name tcolor0_01 \
modis.data ./testdata/input/MYDcrefl.08085185938.hdf \
modis.fireloc ./testdata/input/MYD014.08085185938.txt \
modis.geo ./testdata/input/MYD03.08085185938.hdf \
modis.h2gout ./testdata/output/TCOLOR0.01_subset.tif \
output.type geotiff.rgb \
botlat 35.0 \
toplal 55.0 \
leftlon -105.0 \
rightlon -85.0
```

## **11.2 User-defined Products**

H<sub>2</sub>G is pre-configured to generate standard image products only. To generate user-defined outputs, the user must write a configuration file in the correct format. The user must then set the config.name label to the configuration file path and the config.type label to either 'singleband' or 'rgb'. Detailed instructions for writing configuration files and using them in command lines follow.

### **11.2.1 Configuration Files for User-defined Products**

In order to generate non-standard products the user must create a configuration file describing the desired image product. The configuration file must set values for various configuration keywords. There are two types of configuration files: configuration files for single-band images, and configuration files for RGB images. Descriptions of these two types of configuration files follow.

**NOTE:** Configuration file examples are provided in the testscripts/more\_examples and testscripts/other\_examples directories. Command line examples for using the configuration files are provided in the run-xxx files within these directories. All command lines are commented. Please uncomment the desired command line and then type `./run-xxx` to execute the command. Some H<sub>2</sub>G user-defined products are available for comparison in the testdata/output/more\_examples and testdata/output/other\_examples subdirectories.

#### **11.2.1.1 Configuration Files for User-defined Single-band Images**

##### **Example 7: Single-band configuration file (LST with an NDVI mask)**

The following configuration file 'lstndvimaskconfig.txt' is available in testscripts/more\_examples.

```
DATASET      RR_LST_1KM
GEOLOCATION   MODIS1DB_MOD03_LAT_1KM MODIS1DB_MOD03_LON_1KM
GEOSOURCE     GEOFILE
MASK          RR_NDVI_250M CONTINUOUS 2 -1000 2500
MASKSOURCE    MASKFILE
SCALING       LINEAR 2300 3400
COLORMAP      ./colormap.txt
LEGEND        LST(degK) 5 1 230 63 257.5 128 285 191 312.5 255 340
PROJECTION    GEOGRAPHIC 0.01
SCANS_PER_LOOP 10
```

**NOTE:** There should be no empty lines (even after the last line) in the configuration file.

**Example 8: Running H<sub>2</sub>G using the single-band configuration file** (run from the testscripts/more\_examples directory)

```
$ ../../wrapper/h2g/run \
    config.type singleband \
    config.name ./lstdvimapconfig.txt \
    modis.data ../../testdata/input/LST.08085185938.hdf \
    modis.mask ../../testdata/input/MYD013.08085185938.hdf \
    modis.geo ../../testdata/input/MYD03.08085185938.hdf \
    modis.h2gout ../../testdata/output/LST_NDVIMASK.jpg \
    output.type jpeg.argb
```

A successful execution usually takes some time (approximately 5 minutes, depending on the speed of your computer), so if the execution seems to get stuck, do not become impatient. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be recognized). Correct the problem and run again. The problem can also be identified using the stdfile\* and errfile\* log files. Log files are automatically generated within the directory used for execution.

#### 11.2.1.2 Keyword Descriptions for Single-band User-defined Image Configuration Files

##### **DATASET *DatasetKey***

[Mandatory] The DATASET keyword must be followed by a *DatasetKey*. A *DatasetKey* is an identifier for the dataset for which you want to generate the image. The dataset must be present in the HDF file given to H<sub>2</sub>G via the modis.data command line label. Appendix A lists the DatasetKeys for the datasets and the L2 MODIS products currently supported by H<sub>2</sub>G.

##### **GEOLOCATION *LatitudeDatasetKey* *LongitudeDatasetKey***

[Mandatory] The GEOLOCATION keyword identifies the geolocation dataset that will be used to project the image. Appendix B lists the LatitudeDatasetKeys and LongitudeDatasetKeys, and their HDF sources. Use the MOD03 geolocation keys if the L2 HDF files do not contain any geolocation information (e.g., HDF outputs from the NDVIEVI\_SPA do not contain latitude/longitude datasets). Otherwise use the geolocation datasets that are already included within the modis.data L2 HDF (e.g., HDF outputs from IMAPP\_SPA have their own geolocation datasets).

##### **GEOSOURCE <DATASETFILE | GEOFILE>**

[Mandatory] The GEOSOURCE keyword is used to tell H<sub>2</sub>G if the geolocation datasets are included within the primary L2 HDF dataset or must be extracted from a separate HDF file (like the MOD03 dataset). GEOSOURCE can take one of the two values: DATASETFILE or GEOFILE. If the GEOSOURCE value is

DATASETFILE, H<sub>2</sub>G understands that the geolocation datasets are inside the HDF file identified by the modis.data label. In this case the modis.geo label need not be used on the command line. If the GEOSOURCE value is GEOFILE, H<sub>2</sub>G extracts the geolocation datasets from a separate file identified to it via the modis.geo command line label.

### **MASK DatasetKey <DISCRETE|CONTINUOUS> #NoOfValues #Value1 #Value2**

[Optional] The MASK keyword identifies the dataset used as a mask for the image product. It must be followed by a DatasetKey. (Appendix A lists the DatasetKeys for the datasets and the L2 MODIS products currently supported by H<sub>2</sub>G). The next few values describe how the mask is to be used. The first value after the DatasetKey identifies whether the mask is DISCRETE or CONTINUOUS. A mask is DISCRETE when you want to mask areas where the mask dataset has certain discrete values. A mask is CONTINUOUS when you want to mask areas where the mask dataset values are within a particular continuous range. The next few values identify which values or ranges to mask. Examples to clarify the use of this keyword are provided below.

Example :

MASK RR\_FIREMASK\_1KM DISCRETE 2 3 4

This indicates that the RR\_FIREMASK\_1KM product will be used as a mask. The mask is discrete, meaning some discrete values need to be masked. The '2' tells H<sub>2</sub>G how many values need to be masked; in this case two values of '3' and '4' are to be masked ('3' and '4' represent cloud and water respectively in the RR\_FIREMASK\_1KM dataset).

Example :

MASK RR\_LST\_1KM CONTINUOUS 4 2000 2700 3000 3500

This indicates that the RR\_LST\_1KM product will be used as a mask. The mask is continuous, meaning values within some ranges need to be masked (more than one range can be specified). The first '4' tells H<sub>2</sub>G how many values follow after it. Here LST values in the dataset between two ranges, 2000-2700 and 3000-3500, will be masked. Note that the ranges should be specified in terms of the actual values inside the dataset and not the parameter values. Since the LST dataset appears with a scaling factor of 0.1 inside the HDF file, you should use 2000-2700 to mean 200K-270K.

### **MASKSOURCE <DATASETFILE | MASKFILE>**

[Mandatory when the MASK keyword is used] The MASKSOURCE keyword is used to tell H<sub>2</sub>G if the geolocation datasets are included within the primary L2 HDF dataset, or if they must be extracted from a separate HDF file. MASKSOURCE can take one of the two values, DATASETFILE or MASKFILE. If

the MASKSOURCE value is DATASETFILE, H<sub>2</sub>G understands that the mask dataset is available within the HDF file identified by the modis.data label. In this case the modis.mask label need not be used on the command line. If the MASKSOURCE value is MASKFILE, H<sub>2</sub>G extracts the mask dataset from a separate file identified to it via the modis.mask command line label.

## **FIRERESOURCE FIREFILE**

[Optional] The FIRESOURCE keyword is used to overlay fire pixels on browse images. Note that the FIRESOURCE keyword is ignored when the output.type is geotiff.u8cm. If you use the FIRESOURCE keyword in your configuration file, you must provide the firelocation text file (output from the MOD14\_SPA) as input to H<sub>2</sub>G via the modis.fireloc command line label.

## **SCALING <LINEAR| SEGMENTED\_LINEAR | NONLINEAR> *ScalingParameters***

[Mandatory] The SCALING keyword describes the scaling used to transform the dataset values in the input HDF file into 8-bit values between 0 and 255 in the output image. There are three types of SCALING that can be used. The number and type of *ScalingParameters* differs depending on the type of SCALING employed. SCALING types and examples are provided below.

### **LINEAR *minValue maxValue***

Example:

```
SCALING      LINEAR -1000 10000
```

The HDF values between minValue and maxValue will be scaled linearly between 1 and 255. Any value < minValue will be rounded up to 1. Any value > maxValue will be rounded down to 255. FillValues will be set as 0 (see Appendix A for FillValues for each dataset). In this example values between -1000 and 10000 will be scaled linearly between 1 and 255.

### **SEGMENTED\_LINEAR *noOfTiePoints <hdfValue imageValue>*\***

Example:

```
SCALING      SEGMENTED_LINEAR 3 -1000 1 2500 50 7500 255
```

The SEGMENTED\_LINEAR scaling allows the user to dedicate more color variations within a desired range. In this example, the scale has 3 tie-points and two linear segments. Values between -1000 and 2500 will be scaled linearly between 1 and 50, and values between 2500 and 7500 will be scaled linearly between 50 and 255.

### **NONLINEAR *c0 c1 c2 c3 d***

Example:

```
NONLINEAR -253 0 0 0 63.5
```

The scaling will be done using the following function:

$$\text{geotiff value} = c_0 + c_1 * \text{hdfvalue} + c_2 * \text{hdfvalue}^2 + c_3 * \text{hdfvalue}^3 + d * \ln(\text{hdfvalue})$$

Anything above 255 will be rounded to 255 and any value below 1 will be rounded up to 1.

### **COLORMAP <Standard\_ColorMap | PathToColorMapFile>**

[Mandatory] The COLORMAP keyword identifies an 8-bit colormap for the image product. The colormap has 256 RGB triplets, one for each possible pixel value from 0 to 255. Users can use either a standard colormap by specifying its identifier, or specify a path to their own colormap file. Standard color map identifiers are as follows:

- a) NDVI
- b) LST
- c) FIRE
- d) SST
- e) CHLOR
- f) AOD
- g) IRPHASE
- h) CTP
- i) TPW
- j) CLOUDMASK

A user-defined colormap file must conform to a particular format. An example is contained in Figure 1.

#idx	R	G	B
0	0	0	0
1	144	0	111
2	141	0	114
...	...	...	...
...	...	...	...
...	...	...	...
255	100	0	0

**Figure 1. User-defined Colormap File Example**

**NOTE:** See colormap file format in testscripts/other\_examples/colormap.txt.

## **LEGEND *LegendString* *NoOfTicks* <*TickPosition TickString*>\***

[Optional] The LEGEND keyword can be used to burn a legend into a jpeg or png browse image. Note that the LEGEND keyword is used only when the output.type is either jpeg.argb or png.argb; it is ignored otherwise. The legend shows 256 colors corresponding to the colormap (standard or user-defined).

Example:

```
LEGEND      Cloudtop_Pressure(hPa) 5 1 10 64 274 128 550 192 827  
           255 1100
```

The legend string is Cloudtop\_Pressure(hPa). (Note that no blanks are allowed within the legend string.) The legend shows 255 colors and has 5 ticks specified by the first 5 after the legend string. The string 10 is burned in at position 1, 274 at position 64, and so on. The legend would be burned in at the left bottom corner of the image and would appear as shown in Figure 2.



**Figure 2. Legend Example**

## **PROJECTION GEOGRAPHIC Resolution**

[Mandatory] The PROJECTION keyword specifies the projection and resolution of the output image. Resolution should be specified in degrees. Currently only the 'GEOGRAPHIC' projection is supported within a configuration file. However, you can always override the default projection and resolution on the command line (see paragraph 11.2.2).

**NOTE:** A geographic projection may not be suitable for very high latitudes. Use stereographic projections instead.

## **SCANS\_PER\_LOOP      *NoOfScansProcessedAtATime***

[Mandatory] This is a processing parameter that can be adjusted based on the memory requirements of the output product. For example if memory is low, you can set SCANS\_PER\_LOOP to 10, meaning 10 scans will be processed at a time, thus requiring less memory. Setting SCANS\_PER\_LOOP to low values is useful when generating high-resolution images that require more memory. However, setting SCANS\_PER\_LOOP to low values also slows execution.

## **NO\_INTERPOLATION #*NoOfValues* <*Values*>\***

[Optional] This keyword may be useful when certain values should not be used for interpolation during projection. This keyword's utility is illustrated by the following example. The values '7', '8' and '9' within the RR\_FIREMASK\_1KM dataset in the MOD14 HDF product denote fire pixels. It is desirable that during projection of the image product, these fire pixel values not be used for

interpolation; otherwise, there may be spurious fire pixels in the image product resulting from interpolation.

**Example:**

**NO\_INTERPOLATION 3 7 8 9**

3 values, namely '7', '8' and '9', should not be used for interpolation during creation of the projected image.

### 11.2.1.3 Configuration Files for User-defined RGB Images

#### Example 9: RGB configuration file (False Color Image from the Corrected Reflectance HDF product with fire pixel overlays):

The following configuration file 'fcolorconfig.txt' is available in testscripts/other\_examples.

REDDATASET	CREFL250M_BAND2
REDSCALE	6 0 0 1294 110 2588 160 5176 210 8196 240 11000 255
GREENDATASET	CREFL250M_BAND1
GREENSCALE	6 0 0 1294 110 2588 160 5176 210 8196 240 11000 255
BLUEDATASET	CREFL250M_BAND4
BLUESCALE	6 0 0 1294 110 2588 160 5176 210 8196 240 11000 255
GEOLOCATION	MODISL1DB_MOD03_LAT_1KM MODISL1DB_MOD03_LON_1KM
GEOSOURCE	GEOFILE
FIRESOURCE	FIREFILE
PROJECTION	GEOGRAPHIC 0.01
SCANS_PER_LOOP	10

**NOTE:** There should be no empty lines (even after the last line) in the configuration file.

#### Example 10: Running H<sub>2</sub>G using the RGB configuration file (run from the testscripts/other\_examples directory)

```
$../../wrapper/h2g/run \
  config.type rgb \
  config.name ./fcolorconfig.txt \
  modis.data ../../testdata/input/MYDcrefl.08085185938.hdf \
  modis.fireloc ../../testdata/input/MYD014.08085185938.txt \
  modis.geo ../../testdata/input/MYD03.08085185938.hdf \
  modis.h2gout ../../testdata/output/FCOLOR.jpg \
  output.type jpeg.argb
```

A successful execution usually takes some time (approximately 5 minutes, depending on the speed of your computer), so if the execution seems to get stuck, do not become impatient. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be recognized).

Correct the problem and run again. The problem can also be identified using the stdfile\* and errfile\* log files. Log files are automatically generated within the directory used for execution.

#### 11.2.1.4 Keyword Descriptions for RGB User-defined image Configuration Files

##### **REDDATASET**    *DatasetKey*

[Mandatory] The REDDATASET keyword specifies the dataset used as the Red band in the output RGB image. A *DatasetKey* is an identifier for the dataset you wish to use as the Red band. The dataset must be present in the HDF file given to H<sub>2</sub>G via the modis.data command line label. Appendix A lists the DatasetKeys for the various datasets.

##### **REDSCALE**    *noOfTiePoints [hdfValue integerValue]\**

[Mandatory] The REDSCALE keyword describes the scaling used to transform the red dataset values in the input HDF file into 8-bit values between 0 and 255. A segmented linear scaling is used for RGB images.

Example:

```
REDSCALE    6 0 0 1294 110 2588 160 5176 210 8196 240 11000 255
```

In the above example, the scale has 6 tie-points (and five linear segments) identified by the first 6 after the keyword. Values between 0 and 1294 will be scaled linearly between 0 and 110; values between 1294 and 2588 will be scaled linearly between 110 and 160; values between 2588 and 5176 will be scaled linearly between 160 and 210; values between 5176 and 8196 will be scaled linearly between 210 and 240; and values between 8196 and 11000 will be scaled linearly between 240 and 255.

##### **GREENDATASET**    *DatasetKey*

[Mandatory] The GREENDATASET keyword specifies the dataset used as the Green band in the output RGB image. A *DatasetKey* is an identifier for the dataset you wish to use as the Green band. The dataset must be present in the HDF file given to H<sub>2</sub>G via the modis.data command line label. Appendix A lists the DatasetKeys for the various datasets.

##### **GREENSCALE**    *noOfSegments [hdfValue integerValue]\**

[Mandatory] The GREENSCALE keyword describes the scaling used to transform the green dataset values in the input HDF file into 8-bit values between 0 and 255. A segmented linear scaling is used for RGB images.

##### **BLUEDATASET**    *DatasetKey*

[Mandatory] The BLUEDATASET keyword specifies the dataset used as the Blue band in the output RGB image. A *DatasetKey* is an identifier for the dataset you

want to use as the Blue band. The dataset must be present in the HDF file given to H<sub>2</sub>G via the modis.data command line label. Appendix A lists the DatasetKeys for the various datasets.

**BLUESCALE      *noOfSegments [hdfValue imageValue]*\***

[Mandatory] The BLUESCALE keyword describes the scaling used to transform the blue dataset values in the input HDF file into 8-bit values between 0 and 255. A segmented linear scaling is used for RGB images.

**NOTE:** The REDDATASET, BLUEDATASET and the GREENDATASET must exist in the same HDF input file provided to H<sub>2</sub>G via the modis.data command line label.

**GEOLOCATION LatitudeDatasetKey LongitudeDatasetKey**

[Mandatory] See configuration file entries for single-band images in paragraph 11.2.1.2.

**GEOSOURCE <DATASETFILE | GEOFILE>**

[Mandatory] See configuration file entries for single-band images in paragraph 11.2.1.2.

**FIRESOURCE FIREFILE**

[Optional] See configuration file entries for single-band images in paragraph 11.2.1.2.

**PROJECTION GEOGRAPHIC *Resolution***

[Mandatory] See configuration file entries for single-band images in paragraph 11.2.1.2.

**SCANS\_PER\_LOOP      *NoOfScansProcessedAtATime***

[Mandatory] See configuration file entries for single-band images in paragraph 11.2.1.2.

### **11.2.2 Overriding Projection and Resolution in the User-defined Configuration File**

In order to override the projection specified in the user-defined configuration file you should use the projection and resolution parameter labels on the command line. Note that the geographic projection requires resolution in latitude/longitude degree units, while the stereographic projection requires resolution in meter units. You may also override only the resolution label to modify the resolution of the output image.

**Example 11: Overriding projection and resolution defined in the configuration file (run from the testscripts/more\_examples directory)**

```
$../../wrapper/h2g/run \
    config.type singleband \
    config.name ./lstndvimapconfig.txt \
    modis.data ../../testdata/input/LST.08085185938.hdf \
    modis.mask ../../testdata/input/MYD013.08085185938.hdf \
    modis.geo ../../testdata/input/MYD03.08085185938.hdf \
    modis.h2gout ../../testdata/output/LST_NDVIMASK.jpg \
    output.type jpeg.rgb \
    projection stereographic \
    resolution 1000
```

### 11.2.3 Subsetting User-defined Products

Use the following parameter labels on the command line to specify a region of interest:

- a) botlat (specifies bottom latitude);
- b) toplat (specifies top latitude);
- c) leftlon (specifies left longitude);
- d) rightlon (specifies right longitude).

**NOTE:** Subsetting is currently available for the geographic projection only. If any of these parameters are specified for the stereographic projection, they will be ignored.

**Example 12: Subsetting user-defined products (run from the testscripts/more\_examples directory)**

```
$../../wrapper/h2g/run \
    config.type singleband \
    config.name ./lstndvimapconfig.txt \
    modis.data ../../testdata/input/LST.08085185938.hdf \
    modis.mask ../../testdata/input/MYD013.08085185938.hdf \
    modis.geo ../../testdata/input/MYD03.08085185938.hdf \
    modis.h2gout ../../testdata/output/LST_NDVIMASK_subset.tif \
    output.type geotiff.u8cm \
    botlat 35.0 \
    toplat 55.0 \
    leftlon -105.0 \
    rightlon -85.0
```

## Appendix A

### Dataset Identifiers

Dataset Identifier	Corresponding Scientific Data Set (SDS)	Fill Value	SPA Producing the HDF Product That Contains the SDS
RR_NDVI_250M	NDVI	-999	NDVIEVI_SPA
RR_EVI_250M	EVI	-999	NDVIEVI_SPA
RR_FIREMASK_1KM	fire mask	0	MOD14_SPA
RR_LST_1KM	LST	0	MODLST_SPA
RR_T31_1KM	T31	0	MODLST_SPA
RR_T32_1KM	T32	0	MODLST_SPA
MSL12_SST_1KM	sst	-32767	L2GEN_SPA (SST)
MSL12_QUALSST_1KM	qual_sst	0	L2GEN_SPA (SST)
MSL12_CHLORA_1KM (see Notes below)	chlor_a	-32767 (see Notes below)	L2GEN_SPA (Chlor)
MSL12_L2FLAGS_1KM (see Note below)	l2flags	(see Notes below)	L2GEN_SPA (Chlor)
IMAPP_AOD_10KM	Optical_Depth_Land_And_Ocean	-9999	IMAPP_SPA (mod04)
IMAPP_ODR_10KM	Optical_Depth_Ratio_Small_Land_And_Ocean	-9999	IMAPP_SPA (mod04)
IMAPP_CLOUDPHASE_5KM	Cloud_Phase_Infrared	127	IMAPP_SPA (mod06)
IMAPP_CTP_5KM	Cloud_Top_Pressure	-32768	IMAPP_SPA (mod06)
IMAPP_CTT_5KM	Cloud_Top_Temperature	-32768	IMAPP_SPA (mod06)
IMAPP_CLFR_5KM	Cloud_Fraction	127	IMAPP_SPA (mod06)
IMAPP_CEMS_5KM	Cloud_Effective_Emissivity	127	IMAPP_SPA (mod06)
IMAPP_STYPE_5KM	Surface_Type	-32768	IMAPP_SPA (mod06)
IMAPP_STEMP_5KM	Surface_Temperature	-32768	IMAPP_SPA (mod06)

IMAPP_SPRES_5KM	Surface_Pressure	-32768	IMAPP_SPA (mod06)
IMAPP_SELEV_5KM	Surface_Elevation	-32768	IMAPP_SPA (mod07)
IMAPP_TRHGT_5KM	Tropopause_Height	-32768	IMAPP_SPA (mod06)
IMAPP_TPW_5KM	Water_Vapor	-9999	IMAPP_SPA (mod07)
IMAPP_OZONE_5KM	Total_Ozone	-32768	IMAPP_SPA (mod07)
IMAPP_TOTALS_5KM	Total_Totals	-32768	IMAPP_SPA (mod07)
IMAPP_LIFTEDIDX_5KM	Lifted_Index	-32768	IMAPP_SPA (mod07)
IMAPP_KIDX_5KM	K_Index	-32768	IMAPP_SPA (mod07)
IMAPP_TPWD_5KM	Water_Vapor_Direct	-9999	IMAPP_SPA (mod07)
IMAPP_CLOUDMASK_1KM (see Notes below)	Cloud Mask	0 (see Note below)	IMAPP_SPA (mod35)
CREFL250M_BAND1	CorrRefl_01	32767	CREFL_SPA (default resolution)
CREFL250M_BAND2	CorrRefl_02	32767	CREFL_SPA (default resolution)
CREFL500M_BAND3	CorrRefl_03	32767	CREFL_SPA (default resolution)
CREFL500M_BAND4	CorrRefl_04	32767	CREFL_SPA (default resolution)
CREFL500M_BAND5	CorrRefl_05	32767	CREFL_SPA (default resolution)
CREFL500M_BAND6	CorrRefl_06	32767	CREFL_SPA (default resolution)
CREFL500M_BAND7	CorrRefl_07	32767	CREFL_SPA (default resolution)
MOD09_250MBAND1	Surface Reflectance Band 1	-28672	MOD09_SPA
MOD09_250MBAND2	Surface Reflectance Band 2	-28672	MOD09_SPA
MOD09_500MBAND3	Surface Reflectance Band 3	-28672	MOD09_SPA

MOD09_500MBAND4	Surface Reflectance Band 4	-28672	MOD09_SPA
MOD09_500MBAND5	Surface Reflectance Band 5	-28672	MOD09_SPA
MOD09_500MBAND6	Surface Reflectance Band 6	-28672	MOD09_SPA
MOD09_500MBAND7	Surface Reflectance Band 7	-28672	MOD09_SPA

**Notes on special datasets:** H<sub>2</sub>G reads in the unscaled SDS values from input HDF files. However, there are three special datasets for which H<sub>2</sub>G uses special conversions before any processing. Users should use the appropriate SCALING parameters in configuration files when using them as primary datasets, or use the correct values when using them as masks.

- **MSL12\_CHLORA\_1KM:** The original floating point HDF values are multiplied by 1000000.0 to create integers before H<sub>2</sub>G uses them for further processing.
- **IMAPP\_CLOUDMASK\_1KM:** The original values in HDF are bit fields. They are converted into integer values by H<sub>2</sub>G for convenience. The values are: 0-No Value; 1-Cloudy; 2-Uncertain; 3-Probably Clear; 4-Clear.
- **MSL12\_L2FLAGS\_1KM:** The original values in HDF are bit fields. They are converted into integer values by H<sub>2</sub>G for convenience. The values are: 0-Chlor\_Ok; 1-Chlor\_Warning ; 2-Chlor\_Fail. MSL12\_L2FLAGS\_1KM can only be used as a mask.

## Appendix B

### Geolocation Identifiers

Latitude Dataset Identifier	Longitude Dataset Identifier	SPA Producing the HDF Product That Contains the Scientific Data Set (SDS)
MODISL1DB_MOD03_LAT_1KM	MODISL1DB_MOD03_LON_1KM	MODISL1DB (mod03)
IMAPP_LAT_5KM	IMAPP_LON_5KM	IMAPP_SPA (mod06,mod07)
IMAPP_LAT_10KM	IMAPP_LON_10KM	IMAPP_SPA (mod04)
MOD09_LAT_1KM	MOD09LON1KM	MOD09_SPA

## **Appendix C**

### **Modifying Maximum Java Heap Size**

To increase/decrease maximum Java heap size, cd into algorithm/bin and open the file h2g.sh. Edit the line '-Xmx2g' to the required value. For example, to decrease maximum Java heap size to 1G, edit it to '-Xmx1g'. To increase it to 4G, edit it to '-Xmx4g'.

**CAUTION:** Decreasing Java heap size may cause some high-resolution image generations to fail.

## Appendix D

### H<sub>2</sub>G Standard Product Descriptions

This appendix describes the scaling used to convert Scientific Dataset (SDS) values in the Level 2 SPA HDF products into 8-bit values in the standard GeoTIFF products. Pseudocodes for converting the GeoTIFF values back into actual parameter values are also provided. Please note that the parameter values obtained by inverse scaling GeoTIFF values will not be exactly equal to the parameter values obtained from actual SDS values (from the HDF products), but they should be close. Use of 8-bit integers in our GeoTIFF products may cause loss of precision. Further, values below and above the SDS data range being scaled into 1-255 are set to 1 and 255 respectively in the GeoTIFF output. Interpolation used during re-projection of swath data may also be a source of difference.

#### 1. Aerosol Optical Depth

HDF SDS name: Optical\_Depth\_Land\_And\_Ocean (generated by IMAPP\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0-5000 are scaled linearly to 1-255 in GeoTIFF output. Fill\_Values (-9999) in SDS are set to 0 in geotiff.

```
if hdf_value = -9999
    geotiff_value=0
else
    geotiff_value=1+round((254/5000)*hdf_value) //scale from 1 to 255
    if(geotiff_value<1)
        geotiff_value=1
    endif
    if(geotiff_value>255)
        geotiff_value=255
    endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: 0-5):

```
if geotiff_value>0
    hdf_value=(geotiff_value-1)*5000/254 //scale 1-255 to 0-5000
    parameter_value=hdf_value*0.001 //apply scaling/offset factors
else //geotiff_value=0
    hdf_value=-9999
    parameter_value=NO_RETRIEVAL
end
```

## 2. Cloud Top Pressure

HDF SDS: Cloud\_Top\_Pressure (generated by IMAPP\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 10-11000 are scaled linearly to 1-255 in GeoTIFF output. Fill\_Values (-32768) are set to 0.

```
if hdf_value = -32768
    geotiff_value=0
else
    geotiff_value=1+round((254/10990)*(hdf_value-10)) //scale from 1 to 255
    if(geotiff_value<1)
        geotiff_value=1
    endif
    if(geotiff_value>255)
        geotiff_value=255
    endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: hPa; range: 1-1100 hPa):

```
if geotiff_value>0
    hdf_value=[(geotiff_value-1)*10990/254]+10 //scale 1-255 to 10-11000
    parameter_value=hdf_value*0.1 //apply scaling and offset
else //geotiff_value=0
    hdf_value=-32768
    parameter_value=NO_RETRIEVAL
end
```

## 3. Cloud Phase

HDF SDS: Cloud\_Phase\_Infrared (generated by IMAPP\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0 to 6 are scaled to 1-7 in GeoTIFF output. Fill Values (127) is set to 0.

```
if hdf_value=127
    geotiff_value=0
else
    geotiff_value=1+hdf_value
end
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter flags  
(units: dimensionless):

```
if(geotiff_value>0
    hdf_value=geotiff_value-1
    parameter_flag=hdf_value (clear = 0; water = 1 or 5; ice = 2 or 4; mixed = 3; uncertain = 6)
else
    hdf_value=127
    parameter_flag=NO_RETRIEVAL
end
```

#### 4. Total Precipitable Water

HDF SDS: Water\_Vapor (generated by IMAPP\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0-20000 are scaled linearly to 1-255 in GeoTIFF output. Fill\_Values (-9999) are set to 0.

```
if hdf_value = -9999
    geotiff_value=0
else
    geotiff_value=1+round((254/20000)*hdf_value) //scale from 1 to 255
    if(geotiff_value<1)
        geotiff_value=1
    endif
    if(geotiff_value>255)
        geotiff_value=255
    endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: cm; range: 0-20cm):

```
if geotiff_value>0
    hdf_value=(geotiff_value-1)*20000/254 //scale 1-255 to 0-20000
    parameter_value=hdf_value*0.001 //apply scaling and offset factors as specified in the HDF SDS
else //geotiff_value=0
    hdf_value=-9999
    parameter_value=NO_RETRIEVAL
end
```

## 5. Cloudmask

HDF SDS: Cloud\_Mask (generated by IMAPP\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
Retrieve bit 0 from byte_1 in HDF SDS
if (bit0 = 0)
    geotiff_value=0
else
    Retrieve bits 2 and 1 (bit21) from byte_1 in HDF SDS
    if (bit21=00) //Cloudy
        geotiff_value=1
    elseif (bit21=01) //Uncertain
        geotiff_value=2
    elseif (bit21=10) //Probably Clear
        geotiff_value=3
    elseif (bit21=11) //Clear
        geotiff_value=4
    endif
endif
```

Pseudo-code to convert GeoTIFF values to actual parameter flags (units: dimensionless):

```
if (geotiff_value=1)
    parameterflag=CLOUDY
elseif (geotiff_value=2)
    parameterflag=UNCERTAIN
elseif (geotiff_value=3)
    parameterflag=PROBABLY_CLEAR
elseif (geotiff_value=4)
    parameterflag=CLEAR
elseif (geotiff_value=0)
    parameterflag=NO_RETRIEVAL
endif
```

## 6. NDVI

HDF SDS: NDVI (generated by NDVIEVI\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -1000 to 10000 are scaled linearly to 1-255 in GeoTIFF output. Fill\_Values (-999) are set to 0.

```

if hdf_value = -999
    geotiff_value=0
elseif pixel has CLOUD or WATER (identified using Active Fire HDF product)
    geotiff_value=0
else
    geotiff_value=1+round((254/11000)*(hdf_value+1000)) //scale from 1 to 255
    if(geotiff_value<1)
        geotiff_value=1
    endif
    if(geotiff_value>255)
        geotiff_value=255
    endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: -0.1 to 1.0):

```

if geotiff_value>0
    hdf_value=[(geotiff_value-1)*11000/254]-1000 //scale 1-255 to -1000 to 10000
    parameter_value=hdf_value*0.0001 //apply scaling and offset factors
else //geotiff_value=0
    hdf_value=-999
    parameter_value=NO_RETRIEVAL
end

```

## 7. FIRE

HDF SDS: fire mask (generated by MOD14\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
geotiff_value=hdf_value
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter flags (units: dimensionless):

```

hdf_value=geotiff_value
(Flag interpretation: 0– missing, 1- not processed, 2- not processed, 3- water, 4- cloud, 5-non-fire, 6- unknown, 7- low confidence fire, 8- nominal confidence fire, 9- high confidence fire)

```

## 8. Land Surface Temperature

HDF SDS: LST (generated by MODLST\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 2300-3400 (equivalent to 230K-340K) are scaled linearly to 1-255 in GeoTIFF

output. Fill\_Values (0) are set to 0.

```
if hdf_value = 0
    geotiff_value=0
elseif (pixel has CLOUD or WATER) (identified using Active Fire HDF product)
    geotiff_value=0
else
    geotiff_value=1+round((254/1100)*(hdf_value-2300)) //scale from 1 to 255
    if(geotiff_value<1)
        geotiff_value=1
    endif
    if(geotiff_value>255)
        geotiff_value=255
    endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: K; range: 230K to 340K):

```
if geotiff_value>0
    hdf_value=[(geotiff_value-1)*1100/254]+2300 //scale 1-255 to 2300-3400
    parameter_value=hdf_value*0.1 //apply scaling and offset factors
else //geotiff_value=0
    hdf_value=0
    parameter_value=NO_RETRIEVAL
end
```

## 9. Sea Surface Temperature

HDF SDS: SST (generated by L2GEN\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -400 to 9000 (equivalent to -2°C to 45°C) are scaled linearly to 1-255 in GeoTIFF output. Fill\_Values (-32767) are set to 0.

```

if hdf_value = -32767
    geotiff_value=0
elseif qual_sst>=3 (identified using qual_sst sds)
    geotiff_value=0
else
    geotiff_value=1+round((254/9400)*(hdf_value+400)) //scale from 1 to 255
    if(geotiff_value<1)
        geotiff_value=1
    endif
    if(geotiff_value>255)
        geotiff_value=255
    endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: C; range: -2°C to 45°C):

```

if geotiff_value>0
    hdf_value=[(geotiff_value-1)*9400/254]-400 //scale 1-255 to -400 to 9400
    parameter_value=hdf_value*0.005 //apply scaling and offset factors as specified in the HDF SDS
else //geotiff_value=0
    hdf_value=-32767
    parameter_value=NO_RETRIEVAL
end

```

## 10. Chlorophyll-a Concentration

HDF SDS: CHLOR\_A (generated by L2GEN\_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```

if(hdf_value=-32767 or l2flag = Chl_warn or l2flag=Chl_fail)
    geotiff_value=0
else
    if(hdf_value<0.01)
        hdf_value=0.01
    endif
    if(hdf_value>100)
        hdf_value=100
    endif
    geotiff_value= round(128+(63.5*(log10(hdf_value)))
end

```

Pseudo-code to convert GeotIFF values to HDF SDS values/actual parameter values (units: mg/m^3; range: 0.01 to 100):

```
if (geotiff_value=0)
  hdf_value=-1
  parameter_value=NO_RETRIEVAL
else
  hdf_value=10^[(geotiff_value-128)/63.5]
  parameter_value=hdf_value
end
```

## 11. True Color Images

MODIS corrected reflectances in bands 1, 4 and 3 generated by CREFL\_SPA were used to create the CREFL true color images. Similarly surface reflectances in bands 1, 4 and 3 generated by MOD09\_SPA were used to create MOD09 true color images. The scalings used on the red, green and blue bands to create aesthetically pleasing true color images were inspired by Gumley, Descloires and Schmaltz (2007), "Creating Reprojected True Color MODIS Images: A Tutorial." These standard true color images are available in different resolutions and, optionally, with fire pixel overlays (see Table 2).